

CLAIMS

What is claimed is:

- 1 1. A method for interactively editing a model comprising a first surface, the method
2 being implemented on a computer comprising a processor, a data storage system, at least
3 one input device and at least one output device, the model being stored on a computer-
4 readable media, the method comprising:
5 loading data of the first surface feature from a database stored in the data storage
6 system;
7 creating an aggregate feature for the first surface feature;
8 creating a first graphics object from the aggregate feature;
9 adding the first graphic object to the application scenegraph;
10 creating a geometry object for the aggregate feature;
11 editing the first surface feature in the model;
12 selectively updating the graphics for the model; and
13 removing the first graphics object of the first surface feature from the application
14 scenegraph.

1 2. The method of claim 1, wherein said step of editing comprises:
2 providing an interface;
3 providing an IGM operative with said interface;
4 providing a GQI operative with the IGM;
5 through the interface, selecting an operation to perform on a second surface
6 feature, the interface being constructed and arranged to inform the IGM of the selection;
7 invoking the operation with the GQI;
8 performing at least one callback from the GQI to the IGM during execution of the
9 operation; and
10 updating the graphics object of the model to refresh the output device.

1 3. The method of claim 1, wherein the step of editing is accomplished via irregular
2 space partitioning.

1 4. The method of claim 1, wherein the step of updating the graphics object includes
2 creating the graphics object.

1 5. The method of claim 2, wherein the step of performing the callback causes a
2 change of state for the aggregate feature.

1 6. The method of claim 5, wherein the change of state for the aggregate feature is in
2 a consistency finite state machine.

1 7. The method of claim 6, wherein the consistency finite state machine manages
2 consistency between geometry and graphics.

1 8. The method of claim 5, wherein the step of performing the callback includes
2 invalidating the graphics object for the aggregate feature.

1 9. The method of claim 5, wherein the step of performing the callback includes
2 validating the geometry of the aggregate feature.

1 10. The method of claim 5, wherein the step of performing the callback includes
2 providing a set of objects comprising:

3 a first geometry modeler feature object;

4 a changed geometry object contained in the geometry modeler feature object;

5 a first meta-property attribute object operatively associated with the changed
6 geometry object;

7 a first meta-property object associated with the first meta-property attribute
8 object, the first meta-property object having:

9 a point set preservation property object;

10 a point set preservation property policy object;

11 a cell back pointer property object;

12 an aggregate back pointer property policy object;

13 a geometry cell object associated with the cell back pointer object;

14 an aggregate cell object associated with the geometry cell object; and
15 a display cell graphics object associated with the aggregate cell object;
16 a second meta-property attribute object associated with the geometry modeler
17 feature object; and
18 a second meta-property object associated with the second meta-property attribute
19 object, the second meta-property object having:
20 a second point set preservation property object;
21 a second point set preservation property policy object;
22 a feature back pointer property object;
23 a second aggregate back pointer property policy object;
24 a geometry feature object associated with the feature back pointer property
25 object;
26 an aggregate feature object associated with the geometry feature object;
27 and
28 a display feature graphics object associated with the aggregate feature
29 object.

- 1 11. The method of claim 10, wherein the step of performing the callback includes:
2 if the callback is for a volume object, then performing an edit callback for a
3 volume feature; otherwise performing an edit callback for a surface feature.

1 12. The method of claim 1, wherein the step of updating includes updating graphics
2 for only those cells that have changed geometrically.

1 13. The method of claim 1, wherein the step of updating includes updating graphics
2 for only those features that have changed topologically.

1 14. The method of claim 1, wherein the step of creating graphics for at least one
2 aggregate feature from the model comprises:

3 obtaining all relevant surface features in the model;

4 for each of the relevant surface features:

5 if the surface feature has a graphics object, then updating the
6 graphics object for the surface feature, otherwise creating a graphics
7 object for the surface feature; and

8 adding the graphics object for the surface feature to a surface
9 scenegraph root node.

1 15. The method of claim 14, wherein the step of creating the graphics object
2 comprises:

3 obtaining aggregate objects of all two-dimensional cells for the surface feature;
4 prompting the aggregate objects of each of the cells to obtain at least one valid
5 graphics object and to add the graphics object to a sub-scenagraph of the surface feature;
6 and
7 validating the graphics object of the surface feature.

1 16. The method of claim 15, wherein the step of obtaining valid graphics object for a
2 cell comprises:

3 if the graphics object for the cell does not exist, then creating the graphics object
4 for the cell and validating the graphics; and
5 if the graphics object for the cell exists and is not valid, then updating the graphics
6 object for the cell and validating the graphics.

1 17. The method of claim 1, wherein the step of updating the graphics object includes
2 referencing a visibility finite state machine.

1 18. The method of claim 17, wherein the visibility finite state machine manages the
2 updating of the graphics objects and the updating of the visibility of the graphics objects.

1 19. The method of claim 18, wherein the visibility finite state machine enables the
2 update of only those graphics of objects that are designated as visible.

1 20. The method of claim 18, wherein the step of updating includes:
2 designating a graphics object as visible;
3 checking the validity of the graphics object;
4 if the graphics object is valid, then adding the graphics object to the scenegraph,
5 otherwise, updating the graphics object and then adding the graphics object to the
6 scenegraph.

1 21. The method of claim 1, wherein the step of creating graphics for at least one
2 volume cell from the model comprises:
3 obtaining at least one relevant active volume cell in the model;
4 ensuring that graphics objects of all two-dimensional cells of each volume cell
5 have been created;
6 for each of the two-dimensional cells, obtaining the aggregates and create a new
7 aggregate if the two-dimensional cell aggregate does not exist;
8 ensuring that each two-dimensional cell has valid graphics;
9 creating a graphics object for each of the at least one volume cell;
10 for each volume cell, adding the graphics content of each two-dimensional cells
11 of the volume cell to the sub-scenegraph of the graphics object of the volume cell; and

12 adding the graphics object for each volume cell to the volume scenegraph root
13 node.

1 22. The method of claim 21, wherein an instantiation of a graphics content, of a two-
2 dimensional cell is shared by at least one scenegraph containing a graphics object of a
3 surface feature that has the two-dimensional cell as a child, and at least one of the
4 scenegraphs that contains graphics objects of either one volume cell or two volume cells
5 that have the two-dimensional cell as part of their boundaries.

1 23. The method of claim 6, wherein the step of updating includes:
2 checking the state of each feature in the scenegraph;
3 if the graphics is valid or if the geometry is invalid for the feature, then not
4 updating the graphics object of the feature;
5 if the geometry is valid and the graphics are invalid, then updating the graphics
6 object of the feature.

1 24. The method of claim 6, wherein the step of updating includes:
2 checking the state of each cell in the scenegraph;
3 if the graphics is valid or if the geometry is invalid for the cell, then not updating
4 the graphics object of the cell;
5 if the geometry is valid and the graphics is invalid, then updating the graphics
6 object of the cell.

1 25. The method of claim 11, wherein the step of performing an edit callback for a
2 volume feature comprises:

3 registering the meta-property split callback class method with the geometry
4 modeler interface, the geometry modeler interface constructed and arranged to be
5 invoked when a volume split event occurs;

6 attaching a first meta-property attribute to at least one volume object contained by
7 the volume feature;

8 receiving, from the geometry modeler interface, a callback specifying a first
9 volume object, a second volume affected by a change to the first volume object and the
10 first meta-property attribute;

11 obtaining a pointer value from the first meta-property attribute and de-referencing
12 the pointer value to locate a first meta-property object;

13 invoking a split callback in the first meta-property object with the first meta-
14 property attribute, the first volume object and the second volume object, comprising:

15 obtaining a first point set preservation property instance;

16 obtaining a first point set preservation policy instance from the property
17 instance;

18 initiating a first split callback to the point set preservation policy instance
19 with the point set preservation property instance, the first volume object and the
20 second volume object comprising:

21 obtaining at least one containing feature for the first volume object;
22 and
23 initiating a feature add child update on the geometry modeler
24 interface with the containing feature and the second volume object;
25 obtaining a cell back pointer property instance; and
26 obtaining an aggregate back pointer property policy instance from the cell
27 back pointer property instance; and initiating a second split callback to the
28 aggregate back pointer property policy instance with the cell back pointer
29 property instance comprising:
30 obtaining a volume geometry cell object from the cell back pointer
31 property instance; and
32 initiating a cell split call to the volume geometry cell object,
33 comprising:
34 initiating a call to the volume cell aggregate patron of the
35 volume geometry cell object to invalidate the graphics of the first
36 volume.

1 26. The method of claim 25, wherein the step of performing a feature add child
2 callback for a volume feature comprises:

- 3 registering the meta-property add child callback class method with the geometry
- 4 modeler interface to be invoked when a feature add child event occurs;
- 5 attaching a second meta-property attribute instance to the volume feature;
- 6 receiving from the geometry modeler interface the add child callback specifying
- 7 the volume feature, a volume object and the second meta-property attribute;
- 8 obtaining a pointer value from the second meta-property attribute and de-
- 9 referencing the pointer value to locate a second meta-property object;
- 10 invoking the add child callback in the second meta-property object comprising:
- 11 obtaining a second point set preservation property instance;
- 12 obtaining a second point set preservation property policy instance from the
- 13 second point set preservation property instance; and
- 14 initiating the add child callback method of the second point set policy
- 15 object with the volume feature and the volume geometry object, comprising:
- 16 attaching the point set preservation property to the volume cell;
- 17 obtaining a feature back pointer property instance;
- 18 obtaining a second aggregate back pointer property policy instance
- 19 from the feature back pointer property instance; and
- 20 initiating the add child callback method of the second aggregate
- 21 back pointer property policy instance with the volume feature, the volume

22 geometry object and the volume feature back pointer property,
23 comprising:
24 initiating an add child notify call to the volume feature geometry
25 object identified by the feature back pointer property instance, comprising:
26 initiating a call to the volume feature aggregate patron of
27 the volume feature object to validate the geometry of the volume
28 feature object; and
29 initiating a call to the volume feature aggregate patron of
30 the volume feature object to invalidate the graphics of the volume
31 feature object.

1 27. The method of claim 11, wherein the step of performing an edit callback for a
2 surface feature comprises:
3 registering the meta-property split callback class method with the geometry
4 modeler interface to be invoked when a surface split event occurs;
5 attaching a first meta-property attribute to at least one surface object contained by
6 the surface feature;
7 receiving from the geometry modeler interface a callback specifying a first
8 surface object, a second surface affected by a change to the first surface and the first
9 meta-property attribute;

10 obtaining a pointer value from the first meta-property attribute and de-referencing
11 the pointer value to locate a first meta-property object;
12 invoking a split callback in the first meta-property object with the first surface
13 object, the second surface object and the first meta-property attribute, comprising:
14 obtaining a first point set preservation property instance;
15 obtaining a first point set preservation policy instance from the property
16 instance; and
17 initiating a first split callback to the point set preservation policy instance
18 with the point set preservation property instance, the first surface object and the
19 second surface object comprising:
20 obtaining at least one containing feature for the first surface object;
21 and
22 initiating a feature add child update on the geometry modeler
23 interface with the containing feature and the second surface object;
24 obtaining a cell back pointer property instance;
25 obtaining an aggregate back pointer property policy instance from the cell
26 back pointer property instance; and
27 initiating a second split callback to the aggregate back pointer property
28 policy instance with the cell back pointer property instance comprising:
29 obtaining a surface geometry cell object from the cell back pointer
30 property instance; and

31 initiating a cell split call to the surface geometry cell object,
32 comprising:
33 initiating a call to the surface cell aggregate patron of the
34 surface geometry cell object to invalidate the graphics of the first
35 surface.

1 28. The method of claim 27, wherein the step of performing a feature add callback
2 comprises:
3 registering the meta-property add child callback class method with the geometry
4 modeler interface to be invoked when a feature add child event occurs;
5 attaching a second meta-property attribute instance to the surface feature;
6 receiving from the geometry modeler interface the add child callback specifying
7 the surface feature, a surface object and the second meta-property attribute;
8 obtaining a pointer value from the second meta-property attribute and de-
9 referencing the pointer value to locate a second meta-property object; and
10 invoking the add child callback in the second meta-property object comprising:
11 obtaining a second point set preservation property instance;
12 obtaining a second point set preservation property policy instance from the
13 second point set preservation property instance; and
14 initiating the add child callback method of the second point set policy
15 object with the surface feature and the surface geometry object, comprising:

16 attaching the point set preservation property to the surface cell;
17 obtaining a feature back pointer property instance;
18 obtaining a second aggregate back pointer property policy instance
19 from the feature back pointer property instance; and
20 initiating the add child callback method of the second aggregate
21 back pointer property policy instance with the surface feature, the surface
22 geometry object and the surface feature back pointer property, comprising:
23 initiating an add child notify call to the surface feature
24 geometry object identified by the feature back pointer property
25 instance, comprising:
26 initiating a call to the surface feature aggregate
27 patron of the surface feature object to validate the geometry
28 of the surface feature object; and
29 initiating a call to the surface feature aggregate
30 patron of the surface feature object to invalidate the
31 graphics of the surface feature object.

1 29. The method of claim 2, wherein the step of performing the callback causes a
2 change of state for the cell.

1 30. The method of claim 29, wherein the change of state for the cell is in a
2 consistency finite state machine.

1 31. The method of claim 2, wherein the second surface feature is not contained within
2 the model.

1 32. The method of claim 2, wherein the second surface feature is contained within the
2 model.

1 33. The method of claim 1, wherein the output device is a display.

1 34. The method of claim 11, wherein the step of performing an edit callback for a
2 volume feature comprises:

3 registering the meta-property merge callback class method with the geometry
4 modeler interface to be invoked when a volume merge event occurs;

5 attaching a first meta-property attribute to at least one volume object contained by
6 the volume feature;

7 receiving from the geometry modeler interface a callback specifying a first
8 volume object, a second volume object, a surface object which formerly bounded the first
9 and second volume objects and which has been removed from the model, and a meta-
10 property attribute;

11 obtaining a pointer value from the geometry model attribute and de-referencing
12 the pointer value to locate a first meta-property object; and
13 invoking a merge callback in the first meta-property object, comprising:
14 obtaining a first point set preservation property instance;
15 obtaining a first point set preservation policy instance from the property
16 instance; and
17 initiating a first merge callback to the point set preservation policy
18 instance with the point set preservation property instance, the first volume object,
19 the second volume object, and the surface object, comprising:
20 obtaining at least one containing feature for the first volume object;
21 and
22 initiating a feature remove child update on the geometry modeler
23 interface with the containing feature and the second volume object;
24 obtaining a cell back pointer property instance;
25 obtaining an aggregate back pointer property policy instance from
26 the cell back pointer property instance; and
27 initiating a second merge callback to the aggregate back pointer
28 property policy instance with the cell back pointer property instance, the
29 first volume object, the second volume object, and the surface object,
30 comprising:

31 obtaining a volume geometry cell object from the cell back
32 pointer property instance, and
33 initiating a cell merge call to the volume geometry cell
34 object, comprising:
35 initiating a call to the volume cell aggregate patron of the
36 volume geometry cell object to invalidate the graphics of the first
37 volume.

1 35. The method of claim 34, wherein the step of performing a feature remove child
2 callback comprises:
3 registering the meta-property remove child callback class method with the
4 geometry modeler interface to be invoked when a feature remove child event occurs;
5 attaching a second meta-property attribute instance to the volume feature;
6 receiving from the geometry modeler interface the remove child callback
7 specifying the volume feature, a volume object and the second meta-property attribute;
8 obtaining a pointer value from the second meta-property attribute and de-
9 referencing the pointer value to locate a second meta-property object; and
10 invoking the remove child callback in the second meta-property object
11 comprising:
12 obtaining a second point set preservation property instance;

13 obtaining a second point set preservation property policy instance from the
14 second point set preservation property instance;
15 initiating the remove child callback method of the second point set policy
16 object with the volume feature and the volume geometry object, comprising:
17 removing the point set preservation property from the volume cell;
18 obtaining a feature back pointer property instance;
19 obtaining a second aggregate back pointer property policy instance
20 from the feature back pointer property instance; and
21 initiating the remove child callback method of the second
22 aggregate back pointer property policy instance with the volume feature,
23 the volume geometry object and the volume feature back pointer property,
24 comprising:
25 initiating an remove child notify call to the volume feature
26 geometry object identified by the feature back pointer property
27 instance, comprising:
28 initiating a call to the volume feature aggregate
29 patron of the volume feature object to validate the
30 geometry of the volume feature object; and
31 initiating a call to the volume feature aggregate
32 patron of the volume feature object to invalidate the
33 graphics of the volume feature object.

1 36. The method of claim 11, wherein the step of performing an edit callback for a
2 surface feature comprises:

3 registering the meta-property merge callback class method with the geometry
4 modeler interface to be invoked when a surface merge event occurs;

5 attaching a first meta-property attribute to at least one surface object contained by
6 the surface feature;

7 receiving from the geometry modeler interface a callback specifying a first
8 surface object, a second surface object, and a curve object which formerly bounded the
9 first and second surfaces and which has been removed from the model and a meta-
10 property attribute;

11 obtaining a pointer value from the geometry model attribute and de-referencing
12 the pointer value to locate a first meta-property object; and

13 invoking a merge callback in the first meta-property object, comprising:

14 obtaining a first point set preservation property instance;

15 obtaining a first point set preservation policy instance from the property
16 instance; and

17 initiating a first merge callback to the point set preservation policy
18 instance with the point set preservation property instance, the first surface object,
19 the second surface object, and the curve object, comprising:

20 obtaining at least one containing feature for the first surface object;

21 and

22 initiating a feature remove child update on the geometry modeler
23 interface with the containing feature and the second surface object;
24 obtaining a cell back pointer property instance;
25 obtaining an aggregate back pointer property policy instance from the cell
26 back pointer property instance; and
27 initiating a second merge callback to the aggregate back pointer property
28 policy instance with the cell back pointer property instance, the first surface
29 object, the second surface object, and the curve object comprising:
30 obtaining a surface geometry cell object from the cell back pointer
31 property instance; and
32 initiating a cell merge call to the surface geometry cell object,
33 comprising:
34 initiating a call to the surface cell aggregate patron of the
35 surface geometry cell object to invalidate the graphics of the first
36 surface.

1 37. The method of claim 36, wherein the step of performing a feature remove
2 callback comprises:
3 registering the meta-property remove child callback class method with the
4 geometry modeler interface to be invoked when a feature remove child event occurs;
5 attaching a second meta-property attribute instance to the surface feature;

6 receiving from the geometry modeler interface the remove child callback
7 specifying the surface feature, a surface object and the second meta-property attribute;
8 obtaining a pointer value from the second meta-property attribute and de-
9 referencing the pointer value to locate a second meta-property object; and
10 invoking the remove child callback in the second meta-property object with the
11 second meta-property attribute, the surface feature and the surface geometry
12 object comprising:
13 obtaining a second point set preservation property instance;
14 obtaining a second point set preservation property policy instance from the
15 second point set preservation property instance; and
16 initiating the remove child callback method of the second point set policy
17 object with the surface feature and the surface geometry object, comprising:
18 removing the point set preservation property from the surface cell;
19 obtaining a feature back pointer property instance;
20 obtaining a second aggregate back pointer property policy instance
21 from the feature back pointer property instance; and
22 initiating the remove child callback method of the second
23 aggregate back pointer property policy instance with the surface feature,
24 the surface geometry object and the surface feature back pointer property,
25 comprising:

26 initiating an remove child notify call to the surface feature
27 geometry object identified by the feature back pointer property
28 instance, comprising:
29 initiating a call to the surface feature aggregate
30 patron of the surface feature object to validate the geometry
31 of the surface feature object; and
32 initiating a call to the surface feature aggregate
33 patron of the surface feature object to invalidate the
34 graphics of the surface feature object.

1 38. A computer system for interactively editing a model having a first surface, the
2 computer system further having a processor, a data storage system, at least one input
3 device, and at least one output device, the computer system further having random access
4 memory constructed and arranged to contain an object structure, the object structure
5 comprising:
6 a geometry query interface object, the geometry query interface object having a
7 GQI material property framework object, the GQI material property framework object
8 comprising:
9 a cc_RefObj object;
10 a gmMP object having an IsA relationship with the cc_RefObj object;

11 a gmMPPolicy object having an IsA relationship with the cc_RefObj
12 object; and
13 a gmMPConstant object having an IsA relationship with the gmMP object
14 an interactive geometric modeling object derived from a common model builder
15 object, the interactive geometric modeling object having a relationship with the GQI
16 material property framework object, and an IGM material property framework object, the
17 IGM material property framework object comprising:
18 a gmMPPolyXYZ object having an IsA relationship with the gmMP
19 object;
20 a gmMPZ object having an IsA relationship with the gmMP object;
21 a gmMPTime object having an IsA relationship with the gmMPZ object;
22 a gmMPDepth object having an IsA relationship with the gmMPZ object;
23 a gmMP2DPoly object having an IsA relationship with the gmMP object;
24 a gmMP2DGrid object having an IsA relationship with the gmMP object;
25 a gmMP3dGrid object having an IsA relationship with the gmMP object;
26 a gmMPName object having an IsA relationship with the gmMPConstant
27 object;
28 an mbCellGMReference object having an IsA relationship with the
29 gmMPConstant object;
30 a mbFtrGMReference object having an IsA relationship with the
31 gmMPConstant object;

32 a vspQualityProp object having an IsA relationship with the
33 gmMPConstant object;

34 a vspTransverseIsotropy object having an IsA relationship with the
35 gmMPConstant object; and

36 a fbFtrParameters object having an IsA relationship with the
37 gmMPConstant object;

38 wherein data is processed by the object structure in order to enable a user to edit
39 the model stored in the database.

1 39. The computer system as in claim 38, the IGM material property framework object
2 further comprising:

3 a gmGradientProp object;

4 a vspDensProp object having an IsA relationship with the gmGradProp object;

5 a vspVelPProp object having an IsA relationship with the gmGradProp object;

6 a vspVelSProp object having an IsA relationship with the gmGradProp object;

7 and

8 a gmResistivityProp object having an IsA relationship with the gmGradProp
9 object.

1 40. The computer system as in claim 38, the IGM material property framework object
2 further comprising:
3 a gmMPNamePropertyPolicy object having a composition relationship with the
4 gmMPName object.

1 41. The computer system as in claim 38, the IGM material property framework object
2 further comprising:
3 a gmMPIGMPropertyPolicy object having a composition association with the
4 mbCellGMReference object and a composition relationship with the mbFtrGMReference
5 object.

1 42. The computer system as in claim 38, the GQI material property framework object
2 further comprising:
3 a gmutil_status object;
4 an mv_vt object; and
5 a gm_mp_atoms object.

1 43. The computer system as in claim 38, the GQI material property framework object
2 further comprising:
3 a gmSysRules object having an IsA relationship with the gmMPConstant object;
4 a gmSysPSPPProperty object having an IsA relationship with the gmMPConstant
5 object;

6 a gmUtilFtrBnd object having an IsA relationship with the gmMPConstant object;
7 a gmMPGQIPolicy object having an IsA relationship with the gmMPPolicy
8 object;
9 a gmSysPSPolicy object having an IsA relationship with the gmMPPolicy object
10 and a composition relationship with the gmSysPSPPProperty object;
11 a gmSysRulesPolicy object having an IsA relationship with the gmMPPolicy
12 object and a composition relationship with the gmSysRules object; and
13 a gmMPVolumePropertyPolicy object having an IsA relationship with the
14 gmMPPolicy object and a composition relationship with the gmMPConstant object.

1 44. The computer system as in claim 43, the GQI material property framework object
2 further comprising:
3 a gmUtilFtrBndPolicy object having an IsA relationship with the
4 gmMPGQIPolicy object.

1 45. The computer system as in claim 43, the GQI material property framework object
2 further comprising:
3 a gmMPTopologyTraversalStates object having a composition relationship with
4 the gmMPPolicy object.

1 46. The computer system as in claim 44, wherein a gmMPIGMPPropertyPolicy object
2 has an IsA relationship with the gmUtilFtrBndPolicy object.

1 47. The computer system as in claim 43, wherein a gmMPNamePropertyPolicy object
2 has an IsA relationship with the gmMPVolumePropertyPolicy object.

1 48. The computer system as in claim 39, wherein a gmMPVolumePropertyPolicy
2 object has a first composition relationship with the gmGradientProp object and a second
3 composition relationship with the gmMPZ object.

1 49. The computer system of claim 38, the computer system further comprising:
2 an aqi_Parameter object having a relationship with the gmMP object;
3 a gmMPTopologyTraversalState object having a relationship with the
4 gmMPPolicy object;
5 a first gqi_MetaProperty object having a relationship with the gmMP object;
6 a second gqi_MetaProperty object having a relationship with the gmMP object;
7 a gqi_AttachmentSite object having a relationship with the first gqi_MetaProperty
8 object; and
9 a gqi_Core object having a relationship with the second gqi_MetaProperty object
10 and an IsA relationship with the gqi_AttachmentSite object.

1 50. The computer system of claim 49, the computer system further comprising:
2 an ag_BaseClass object;
3 an ag_Geometry object having an IsA relationship with the ag_BaseClass object;
4 a gm_Geometry object having an hasA relationship with the ag_Geometry object,
5 the gm_Geometry object having a relationship with the Geometry object;
6 an oi_Feature object having a relationship with an ag_Feature object, the
7 ag_Feature object having an IsA relationship with the ag_Geometry object, the
8 ag_Feature object further having a relationship with the gm_Feature object;
9 an oi_Cell object having a relationship with an ag_Cell object, the ag_Cell object
10 having an IsA relationship with the ag_Cell object, the ag_Cell object further having a
11 relationship with the gm_Cell object;
12 an X_Feature object having a relationship with the gm_Feature object, the
13 X_Feature object having an IsA relationship with an xAttrFtr object, the xAttrFtr object
14 having an IsA relationship with a gqi_Feature object, and the gqi_Feature object having
15 an IsA relationship with the X_Feature object;
16 an X_Cell object, the X_Cell object having an IsA relationship with the
17 X_Feature object, the X_Cell object having an IsA relationship with a first xAttrMP
18 object, the first xAttrMP object having an IsA relationship with a first gqi_MetaProperty
19 object, the first gqi_MetaProperty object having an association with a FtrRef object, the
20 FtrRef object having at least one association with the gm_Feature object; and

21 the X_Feature object further having an IsA relationship with a second xAttrMP
22 object, the second xAttrMP object having an IsA relationship with a second
23 gqi_MetaProperty object, the second gqi_MetaProperty object having an association with
24 CellRef object, the CellRef object having at least one association with the gm_Cell
25 object.

1 51. A random access memory, the random access memory having an object structure
2 comprising:

3 a geometry query interface object, the geometry query interface object having a
4 GQI material property framework object, the GQI material property framework object
5 comprising:

6 a cc_RefObj object;

7 a gmMP object having an IsA relationship with the cc_RefObj object;

8 a gmMPPolicy object having an IsA relationship with the cc_RefObj
9 object; and

10 a gmMPConstant object having an IsA relationship with the gmMP object

11 an interactive geometric modeling object derived from a common model builder
12 object, the interactive geometric modeling object having a relationship with the GQI
13 material property framework object, and an IGM material property framework object, the
14 IGM material property framework object comprising:

15 a gmMPPolyXYZ object having an IsA relationship with the gmMP
16 object;
17 a gmMPZ object having an IsA relationship with the gmMP object;
18 a gmMPTime object having an IsA relationship with the gmMPZ object;
19 a gmMPDepth object having an IsA relationship with the gmMPZ object;
20 a gmMP2DPoly object having an IsA relationship with the gmMP object;
21 a gmMP2DGrid object having an IsA relationship with the gmMP object;
22 a gmMP3dGrid object having an IsA relationship with the gmMP object;
23 a gmMPName object having an IsA relationship with the gmMPConstant
24 object;
25 an mbCellGMReference object having an IsA relationship with the
26 gmMPConstant object;
27 a mbFtrGMReference object having an IsA relationship with the
28 gmMPConstant object;
29 a vspQualityProp object having an IsA relationship with the
30 gmMPConstant object;
31 a vspTransverseIsotropy object having an IsA relationship with the
32 gmMPConstant object; and
33 a fbFtrParameters object having an IsA relationship with the
34 gmMPConstant object;

35 wherein data is processed by the object structure in order to enable a user to edit
36 the model stored in the database.

1 52. The random access memory as in claim 51, the IGM material property framework
2 object further comprising:

3 a gmGradientProp object;

4 a vspDensProp object having an IsA relationship with the gmGradProp object;

5 a vspVelPProp object having an IsA relationship with the gmGradProp object;

6 a vspVelSProp object having an IsA relationship with the gmGradProp object;

7 and

8 a gmResistivityProp object having an IsA relationship with the gmGradProp
9 object.

1 53. The random access memory as in claim 51, the IGM material property framework
2 object further comprising:

3 a gmMPNamePropertyPolicy object having a composition relationship with the
4 gmMPName object.

1 54. The random access memory as in claim 51, the IGM material property framework
2 object further comprising:

3 a gmMPIGMPropertyPolicy object having a composition association with the
4 mbCellGMReference object and a composition relationship with the mbFtrGMReference
5 object.

1 55. The random access memory as in claim 51, the GQI material property framework
2 object further comprising:

3 a gmutil_status object;
4 an mv_vt object; and
5 a gm_mp_atoms object.

1 56. The random access memory as in claim 51, the GQI material property framework
2 object further comprising:

3 a gmSysRules object having an IsA relationship with the gmMPConstant object;
4 a gmSysPSPPProperty object having an IsA relationship with the gmMPConstant
5 object;
6 a gmUtilFtrBnd object having an IsA relationship with the gmMPConstant object;
7 a gmMPGQIPolicy object having an IsA relationship with the gmMPPolicy
8 object;

9 a gmSysPSPolicy object having an IsA relationship with the gmMPPolicy object
10 and a composition relationship with the gmSysPSPProperty object;
11 a gmSysRulesPolicy object having an IsA relationship with the gmMPPolicy
12 object and a composition relationship with the gmSysRules object; and
13 a gmMPVolumePropertyPolicy object having an IsA relationship with the
14 gmMPPolicy object and a composition relationship with the gmMPConstant object.

1 57. The random access memory as in claim 56, the GQI material property framework
2 object further comprising:

3 a gmUtilFtrBndPolicy object having an IsA relationship with the
4 gmMPGQIPolicy object.

1 58. The random access memory as in claim 56, the GQI material property framework
2 object further comprising:

3 a gmMPTopologyTraversalStates object having a composition relationship with
4 the gmMPPolicy object.

1 59. The random access memory as in claim 57, wherein a gmMPIGMPropertyPolicy
2 object has an IsA relationship with the gmUtilFtrBndPolicy object.

1 60. The random access memory as in claim 56, wherein a gmMPNamePropertyPolicy
2 object has an IsA relationship with the gmMPVolumePropertyPolicy object.

1 61. The random access memory as in claim 52, wherein a
2 gmMPVolumePropertyPolicy object has a first composition relationship with the
3 gmGradientProp object and a second composition relationship with the gmMPZ object.

1 62. The random access memory of claim 51, the random access memory further
2 comprising:

3 an aqi_Parameter object having a relationship with the gmMP object;

4 a gmMPTopologyTraversalState object having a relationship with the
5 gmMPPolicy object;

6 a first gqi_MetaProperty object having a relationship with the gmMP object;

7 a second gqi_MetaProperty object having a relationship with the gmMP object;

8 a gqi_AttachmentSite object having a relationship with the first gqi_MetaProperty
9 object; and

10 a gqi_Core object having a relationship with the second gqi_MetaProperty object
11 and an IsA relationship with the gqi_AttachmentSite object.

1 63. The random access memory of claim 62, the random access memory further
2 comprising:

3 an ag_BaseClass object;

4 an ag_Geometry object having an IsA relationship with the ag_BaseClass object;

5 a gm_Geometry object having an hasA relationship with the ag_Geometry object,
6 the gm_Geometry object having a relationship with the Geometry object;

7 an oi_Feature object having a relationship with an ag_Feature object, the
8 ag_Feature object having an IsA relationship with the ag_Geometry object, the
9 ag_Feature object further having a relationship with the gm_Feature object;

10 an oi_Cell object having a relationship with an ag_Cell object, the ag_Cell object
11 having an IsA relationship with the ag_Cell object, the ag_Cell object further having a
12 relationship with the gm_Cell object;

13 an X_Feature object having a relationship with the gm_Feature object, the
14 X_Feature object having an IsA relationship with an xAttrFtr object, the xAttrFtr object
15 having an IsA relationship with a gqi_Feature object, and the gqi_Feature object having
16 an IsA relationship with the X_Feature object;

17 an X_Cell object, the X_Cell object having an IsA relationship with the
18 X_Feature object, the X_Cell object having an IsA relationship with a first xAttrMP
19 object, the first xAttrMP object having an IsA relationship with a first gqi_MetaProperty
20 object, the first gqi_MetaProperty object having an association with a FtrRef object, the
21 FtrRef object having at least one association with the gm_Feature object; and

22 the X_Feature object further having an IsA relationship with a second xAttrMP
23 object, the second xAttrMP object having an IsA relationship with a second
24 gqi_MetaProperty object, the second gqi_MetaProperty object having an association with

- 25 CellRef object, the CellRef object having at least one association with the gm_Cell
- 26 object.

11/11/2011 10:11:11 AM